

Hacia una Integración de Argumentación Rebatible y Ontologías en la Web Semántica

Sergio Alejandro Gómez, Carlos Iván Chesñevar, Guillermo Ricardo Simari

Laboratorio de Investigación y Desarrollo en Inteligencia Artificial (LIDIA)¹

Depto. Cs. e Ing. de la Computación – Universidad Nacional del Sur

Av. Alem 1253 - (8000) Bahía Blanca - Argentina

Tel/Fax: (+54) 291-4595135/6

E-mail: {sag, cic, grs}@cs.uns.edu.ar

Resumen

La World Wide Web actual está basada principalmente en documentos escritos con un énfasis en su presentación visual. La *Web Semántica* es una visión futura de la Web en la cual la información tiene un significado exacto otorgado por medio de *definiciones de ontologías*. En los estándares propuestos por la W3C, no es posible razonar en presencia de información incompleta y potencialmente contradictoria.

Como parte de esta línea de investigación, estamos trabajando en la aplicación de la argumentación rebatible al problema de la representación de ontologías en el contexto de la Web Semántica.

Palabras clave: Argumentación rebatible, Web Semántica, Representación de ontologías.

1. Introducción y motivaciones

La World Wide Web actual está basada principalmente en documentos escritos con un énfasis en su presentación visual teniendo en cuenta a usuarios humanos y no a los programas de computadora. La *Web Semántica* [BLHL01] es una visión futura de la Web en la cual la información tiene un significado exacto; permitiendo así que las computadoras entiendan y razonen en base a la información encontrada en la Web.

La Web Semántica propone resolver el problema de la asignación de semántica a los recursos web por medio de *definiciones de ontologías*. En el contexto del área de compartir conocimiento, el término *ontología* se refiere a la especificación de una conceptualización. Esto es, una ontología es una descripción de los conceptos y relaciones que pueden existir para un agente o una comunidad de agentes [Gru93].

En los enfoques actuales, las ontologías son descriptas en un *lenguaje de representación de ontologías*. El *Lenguaje de Ontologías Web* (OWL) [MvH04] es el lenguaje propuesto por el estándar del Consorcio de la W3C para tal fin. Aunque dicho lenguaje provee un número expresivo de constructores, el mismo adolece del problema de no permitir razonar en presencia de información incompleta y potencialmente contradictoria.

Por lo tanto, como parte de esta línea de investigación, estamos trabajando en la aplicación de la argumentación rebatible al problema de la representación de ontologías en el contexto de la Web Semántica.

¹LIDIA es un miembro del IICyTI (Instituto de Investigación en Ciencia y Tecnología Informática).

2. Marco de trabajo: Argumentación Rebatible

La *Programación en Lógica Rebatible* (DeLP) [GS04] provee un lenguaje para la representación de conocimiento y el razonamiento que utiliza la *argumentación rebatible* [CML00, PV02, SL92] para decidir entre conclusiones contradictorias a través de un *análisis dialéctico*. La codificación de conocimiento por medio de un programa DeLP provee un buen balance entre expresividad e implementabilidad. La investigación reciente ha mostrado que DeLP provee un marco apropiado para construir aplicaciones del mundo real (e.g., algoritmos de agrupación [GC04] y formularios web inteligentes [GCS05, GCS07]) que pueden lidiar con información incompleta y potencialmente contradictoria.

En un programa lógico rebatible $\mathcal{P} = (\Pi, \Delta)$, se pueden distinguir un conjunto Δ de reglas rebatibles $P \multimap Q_1, \dots, Q_n$, y un conjunto Π de reglas estrictas $P \leftarrow Q_1, \dots, Q_n$. La derivación de literales en DeLP resulta en la construcción de argumentos. Un *argumento* $\langle \mathcal{A}, H \rangle$ es un conjunto no contradictorio y minimal de cláusulas fijas \mathcal{A} de Δ que permite derivar un literal fijo H posiblemente usando reglas fijas de Π .

Debido a que los argumentos pueden estar en conflicto (concepto capturado en términos de una contradicción lógica) puede definirse una relación de ataque entre argumentos. Usualmente se define un criterio para decidir entre dos argumentos en conflicto. Si el argumento atacante es estrictamente preferido sobre el argumento atacado, entonces éste es llamado un *derrotador propio*; en cambio, si no hay comparación posible, entonces éste es llamado un *derrotador por bloqueo*.

Para determinar si un argumento dado \mathcal{A} es considerado finalmente no derrotado (o *garantizado*), se lleva a cabo un proceso dialéctico recursivo, en el cual son tomados en cuenta los derrotadores para \mathcal{A} , los derrotadores de éstos y así sucesivamente. Dado un programa DeLP \mathcal{P} y una consulta H , la respuesta final a H con respecto a \mathcal{P} toma en cuenta tal análisis dialéctico. La respuesta a una consulta puede ser: *sí*, *no*, *indeciso*, o *desconocido*.

3. Trabajo en progreso: Hacia las δ -ontologías

Considerando las dificultades de los enfoques actuales presentadas en la Sección 1, estamos trabajando en un formalismo alternativo de representación de ontologías basado en argumentación rebatible. A continuación, presentamos algunos de los resultados obtenidos.

Intuitivamente, una ontología es una colección de información, incluyendo generalmente información acerca de clases y propiedades. Plantearemos una definición formal de ontología y veremos cómo representar, por cuestiones de espacio, unos pocos ejemplos en este formalismo. Luego veremos cómo a esta definición de ontología puede asignarsele semántica a través de un programa DeLP.

Definición 3.1 (δ -Ontología) Una δ -ontología \mathcal{O} es una 8-upla:

$$\mathcal{O} = \langle \mathcal{H}_D, \mathcal{H}_S, \mathcal{A}_D, \mathcal{A}_S, \mathcal{C}_D, \mathcal{C}_S, \mathcal{I}, \mathcal{R} \rangle$$

donde: \mathcal{H}_D es un conjunto de reglas rebatibles para establecer la jerarquía de clases; \mathcal{H}_S es un conjunto de reglas estrictas para establecer la jerarquía de clases; \mathcal{A}_D es un conjunto de reglas rebatibles para establecer atributos de clases; \mathcal{A}_S es un conjunto de reglas estrictas para establecer atributos de clases; \mathcal{C}_D es un conjunto de reglas rebatibles para establecer restricciones sobre clases e instancias; \mathcal{C}_S es un conjunto de reglas estrictas para establecer restricciones sobre clases e instancias; \mathcal{I} es un conjunto de hechos para definir individuos de clases; y, \mathcal{R} es un conjunto de relaciones entre individuos.

Definición de clases y subclases

Las *clases* corresponden a conjuntos de individuos y se definen mediante predicados unarios. Es decir, una clase C se representará con un predicado unario $C(X)$ ². La noción de *subclase* relaciona una clase S más específica con una clase C . En nuestro acercamiento, consideraremos dos nociones de subclase: *subclase estricta* (representada con una regla estricta $C(X) \leftarrow S(X)$) y *subclase rebatible* (representada con una regla rebatible $C(X) \multimap S(X)$). Es necesario notar que la decisión de cuál tipo de noción de subclase es necesario utilizar es puramente una decisión del ingeniero de conocimiento.

Ejemplo 3.1 La idea que expresa que “absolutamente todos los individuos de la clase “estudiante de doctorado son individuos de la clase persona” (i.e., la clase PhDStudent es subclase estricta de la clase Student), se representará con una regla estricta en el conjunto \mathcal{H}_S :

$$Student(X) \leftarrow PhDStudent(X).$$

En cambio, la idea que establece que “usualmente todos los individuos de la clase estudiante de doctorado son individuos de la clase de individuos de la clase persona insolvente económicamente” (i.e., la clase PhDStudent es subclase rebatible de la clase NonSolvent), se representará con una regla rebatible en el conjunto \mathcal{H}_D :

$$NonSolvent(X) \multimap PhDStudent(X).$$

Definición de individuos

La noción de *instancia* es la relación de pertenencia de un individuo a una clase. En nuestro acercamiento, la noción de individuo perteneciente a una clase está dada por un conjunto de hechos formado por predicados unarios fijos donde el nombre del predicado denota una clase y su argumento corresponde a una constante representando a un individuo en dicha clase.

Ejemplo 3.2 Para representar la información que expresa que John es un estudiante de doctorado y que Mark es millonario usaremos los hechos pertenecientes al conjunto \mathcal{I} :

$$\begin{aligned} & PhDStudent(john) \\ & Millionaire(mark) \end{aligned}$$

Definición de relaciones entre individuos

Para denotar *relaciones* entre dos individuos de dos clases diferentes utilizaremos hechos binarios en el conjunto \mathcal{R} .

Ejemplo 3.3 Para establecer que el padre de John es Eddie, usaremos el siguiente hecho en el conjunto \mathcal{R} :

$$Father(john, eddie)$$

El enfoque a la definición de ontologías que proponemos no limita el uso de relaciones a relaciones binarias; por el contrario, podemos utilizar relaciones n -arias para modelar relaciones generales.

Ejemplo 3.4 Para establecer que un banco posee la información concerniente a que John vive en Krakosia y tiene un salario de \$400 mensuales, tendremos el hecho en \mathcal{R} :

$$Info(john, krakosia, 400)$$

²En este aspecto, nuestro enfoque coincide con los enfoques basados en las Lógicas para la Descripción [BCM⁺03].

Definición de atributos

El mundo de clases e individuos es muy limitado si sólo se pueden definir taxonomías. Los *atributos* permiten especificar hechos generales sobre los miembros de las clases y hechos específicos sobre los individuos. Separaremos los atributos en estrictos y rebatibles. También consideraremos el uso de reglas para especificar chequeo de tipos.

Ejemplo 3.5 *La clase Persona posee los atributos nombre, edad y sexo de tipo string, entero y string respectivamente. Esta información se representa con las siguientes reglas en el conjunto \mathcal{A}_S :*

$$\begin{aligned} \text{string}(Y) &\leftarrow \text{Person}(X), \text{name}(X, Y) \\ \text{integer}(Y) &\leftarrow \text{Person}(X), \text{age}(X, Y) \\ \text{string}(Y) &\leftarrow \text{Person}(X), \text{sex}(X, Y) \end{aligned}$$

También consideraremos *atributos emergentes* de la ontología que proveen conclusiones plausibles en presencia de información incompleta y potencialmente contradictoria. Como el valor de tales atributos puede cambiar en presencia de nuevo conocimiento, los llamaremos *atributos rebatibles* y éstos serán expresados como reglas en el conjunto \mathcal{A}_D .

Ejemplo 3.6 *Para expresar que un estudiante de doctorado usualmente tiene un estado económico poco líquido a menos que su padre sea rico, usaremos las reglas en \mathcal{A}_D :*

$$\begin{aligned} \text{Status}(X, \text{non_liquid}) &\multimap \text{PhDStudent}(X) \\ \text{Status}(X, \text{liquid}) &\multimap \text{PhDStudent}(X), \text{Father}(X, Y), \text{Rich}(Y). \end{aligned}$$

Definición de restricciones sobre clases e instancias

Una vez definidos las clases, los individuos, sus atributos y las relaciones entre individuos de clase, nos interesa poder especificar *restricciones* que permitan razonar sobre ellos y así enriquecer las conclusiones que se pueden obtener sobre los datos. Para ello, utilizaremos dos conjuntos \mathcal{C}_D y \mathcal{C}_S especificando restricciones rebatibles y estrictas respectivamente.

Ejemplo 3.7 *Para indicar que las nociones de liquidez e iliquidez son opuestas, escribiremos en el conjunto \mathcal{C}_S :*

$$\sim \text{Status}(X, \text{liquid}) \leftarrow \text{Status}(X, \text{non_liquid})$$

Para indicar que una persona que no es solvente usualmente no tiene un buen ingreso, utilizaremos la regla en \mathcal{C}_D :

$$\sim \text{Good_income}(X) \multimap \sim \text{Solvent}(X)$$

Razonamiento con δ -ontologías

Dada una definición de una ontología, nos interesa poder razonar sobre su contenido. En virtud de que la información contenida en la misma puede ser incompleta y potencialmente contradictoria, realizaremos un análisis dialéctico sobre las reglas y hechos que la conforman.

Definición 3.2 (Semántica de una δ -ontología) *Una δ -ontología $\mathcal{O} = \langle \mathcal{H}_D, \mathcal{H}_S, \mathcal{A}_S, \mathcal{A}_D, \mathcal{C}_D, \mathcal{C}_S, \mathcal{I}, \mathcal{R} \rangle$ se corresponderá con un programa lógico rebatible:*

$$\mathcal{P} = (\mathcal{H}_S \cup \mathcal{A}_S \cup \mathcal{C}_S \cup \mathcal{I} \cup \mathcal{R}, \mathcal{H}_D \cup \mathcal{A}_D \cup \mathcal{C}_D)$$

Entonces, dada una definición de una ontología \mathcal{O} , nos interesarán todos los argumentos que se pueden obtener a partir del programa DeLP \mathcal{P} que le da semántica. Estos argumentos serán utilizados luego en un análisis dialéctico para obtener las conclusiones garantizadas a partir de la ontología.

Agradecimientos

Esta investigación está financiada por la Agencia Nacional de Promoción Científica y Tecnológica (PICT 2002 No. 13.096, PICT 2003 No. 15.043, PAV 2004 076), por Proyecto 24/ZN10 (Secretaría de Ciencia y Tecnología, Universidad Nacional del Sur, Argentina) y por CONICET (Argentina).

Referencias

- [BCM⁺03] Franz Baader, Diego Calvanese, Deborah McGuinness, Daniele Nardi, and Peter Patel-Schneider, editors. *The Description Logic Handbook – Theory, Implementation and Applications*. Cambridge University Press, 2003.
- [BLHL01] Tim Berners-Lee, James Hendler, and Ora Lassila. The Semantic Web. *Scientific American*, 17 May 2001.
- [CML00] C. Chesñevar, A. Maguitman, and R. Loui. Logical Models of Argument. *ACM Computing Surveys*, 32(4):337–383, December 2000.
- [GC04] S. Gómez and C. Chesñevar. A Hybrid Approach to Pattern Classification Using Neural Networks and Defeasible Argumentation. In *Proc. of 17th Intl. FLAIRS Conference. Miami, Florida, USA*, pages 393–398. American Assoc. for Art. Intel., May 2004.
- [GCS05] Sergio Alejandro Gómez, Carlos Iván Chesñevar, and Guillermo Ricardo Simari. Incorporating Defeasible Knowledge and Argumentative Reasoning in Web-based Forms. *Proc. of the 3rd Intl. Workshop on Intelligent Techniques for Web Personalization (ITWP 2005). In 19th Intl. Joint Conf. in Artificial Intelligence (IJCAI 2005)*, pages 9–16, 2005.
- [GCS07] Sergio Alejandro Gómez, Carlos Iván Chesñevar, and Guillermo Ricardo Simari. Defeasible Reasoning in Web-based Forms through Argumentation. *International Journal of Information Technology & Decision Making (in press)*, 2007.
- [Gru93] T. R. Gruber. A translation approach to portable ontologies. *Knowledge Acquisition*, 5(2):199–220, 1993.
- [GS04] A. García and G. Simari. Defeasible Logic Programming an Argumentative Approach. *Theory and Prac. of Logic Program.*, 4(1):95–138, 2004.
- [MvH04] Deborah L. McGuinness and Frank van Harmelen. OWL Web Ontology Language Overview, 2004. <http://www.w3.org/TR/owl-features/>.
- [PV02] Henry Prakken and Gerard Vreeswijk. Logical Systems for Defeasible Argumentation. In D. Gabbay and F. Guenther, editors, *Handbook of Philosophical Logic*, pages 219–318. Kluwer Academic Publishers, 2002.
- [SL92] G. Simari and R. Loui. A Mathematical Treatment of Defeasible Reasoning and its Implementation. *Artificial Intelligence*, 53:125–157, 1992.